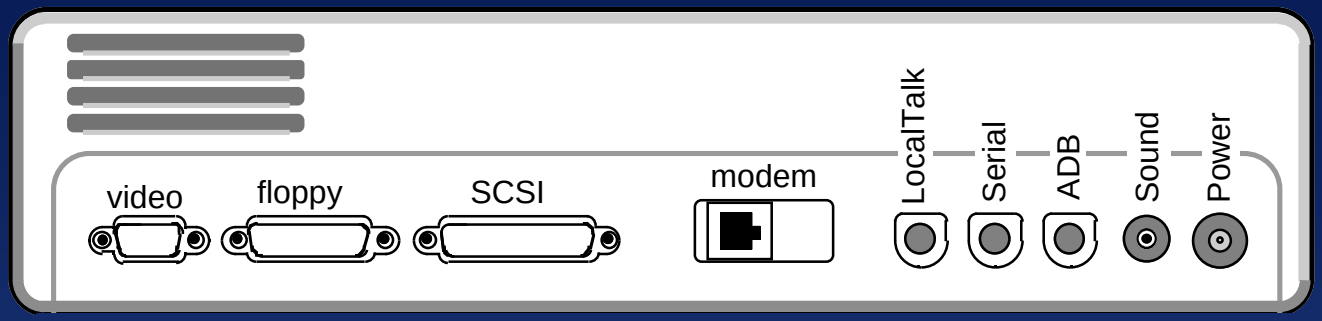
IEEE 1394-1995
High Performance Serial Bus

Michael D. Johas Teener
Editor, P1394 Working Group
Plumbing Architect,
Apple Computer, Inc.
One Infinite Loop, MS 35-OEM
Cupertino, CA 95014
teener@apple.com

Background
(the way things are now)



- No I/O Integration
  - lots of PCB area, silicon & software
  - no common architecture
- Hard to change
  - no realtime transport
  - performance not scalable

Goals

- Low cost, high performance ergonomic peripheral bus
- Read/write memory architecture
  - NOT an I/O channel
- Compatible architecture with other IEEE busses
  - Follow IEEE 1212 CSR (Control and Status Register) standard
- Isochronous service

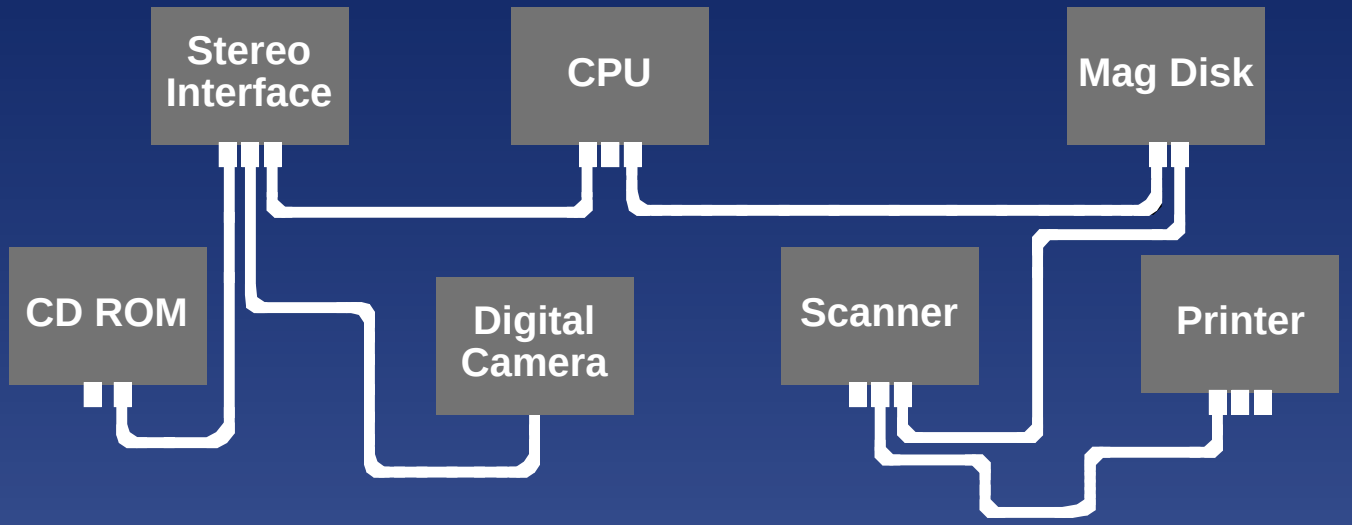
“Isochronous” ??

- **Iso (same) chronous (time) :**
  - **Uniform in time**
  - **Having equal duration**
  - **Recurring at regular intervals**

Data Type	Sample size & rate	Bit rate
ISDN	8 kHz x 8 bits	64 kbps
CD	44.1 kHz x 16 bits x 2 channels	1.4 Mbps
DAT	48 kHz x 16 bits x 2 channels	1.5 Mbps
Video	variable to 30 fps	1.5 – 216 Mbps

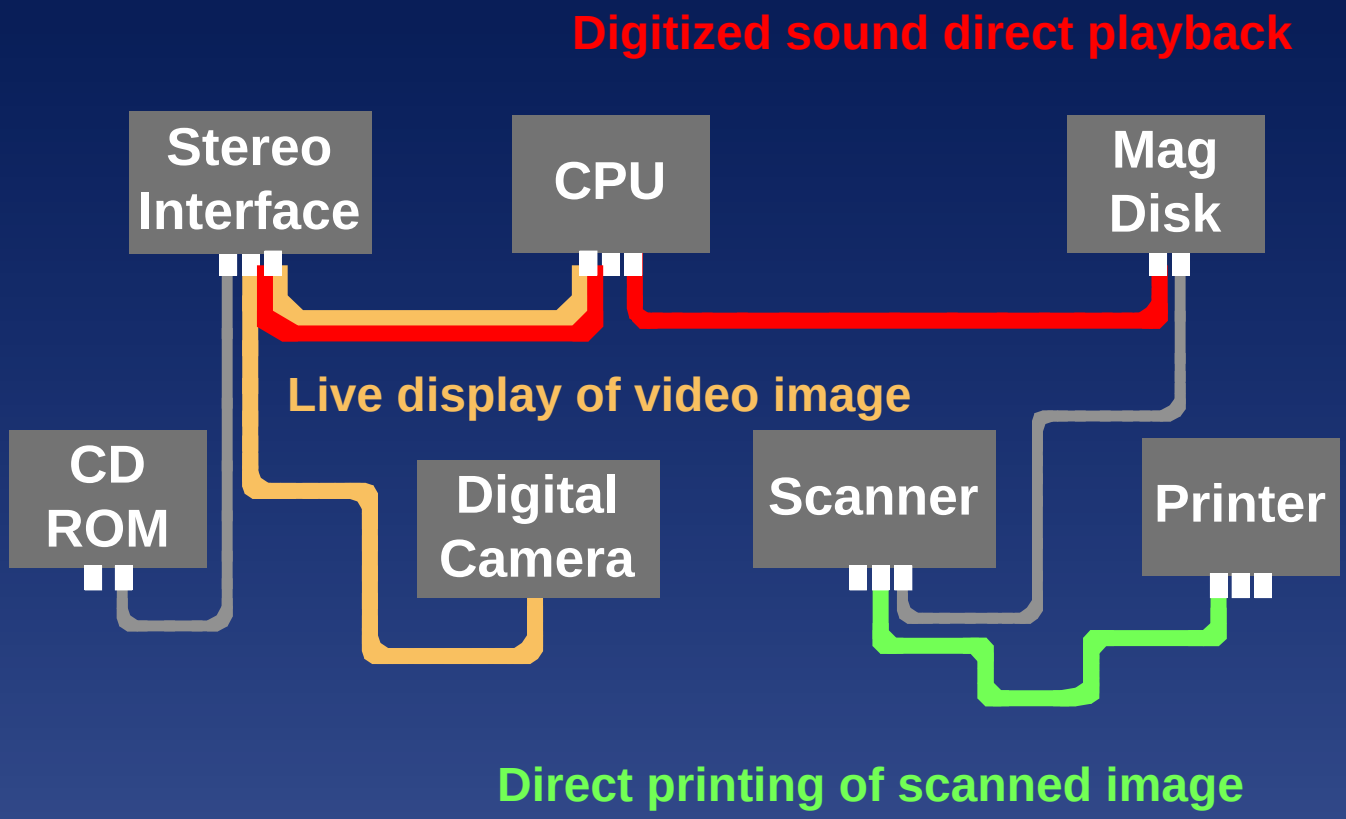
Unsupervised!

SCSI is typical *"supervised cabling"* — daisy-chain; manual or fixed addresses; terminators at ends; devices with internal terminations must be at one end

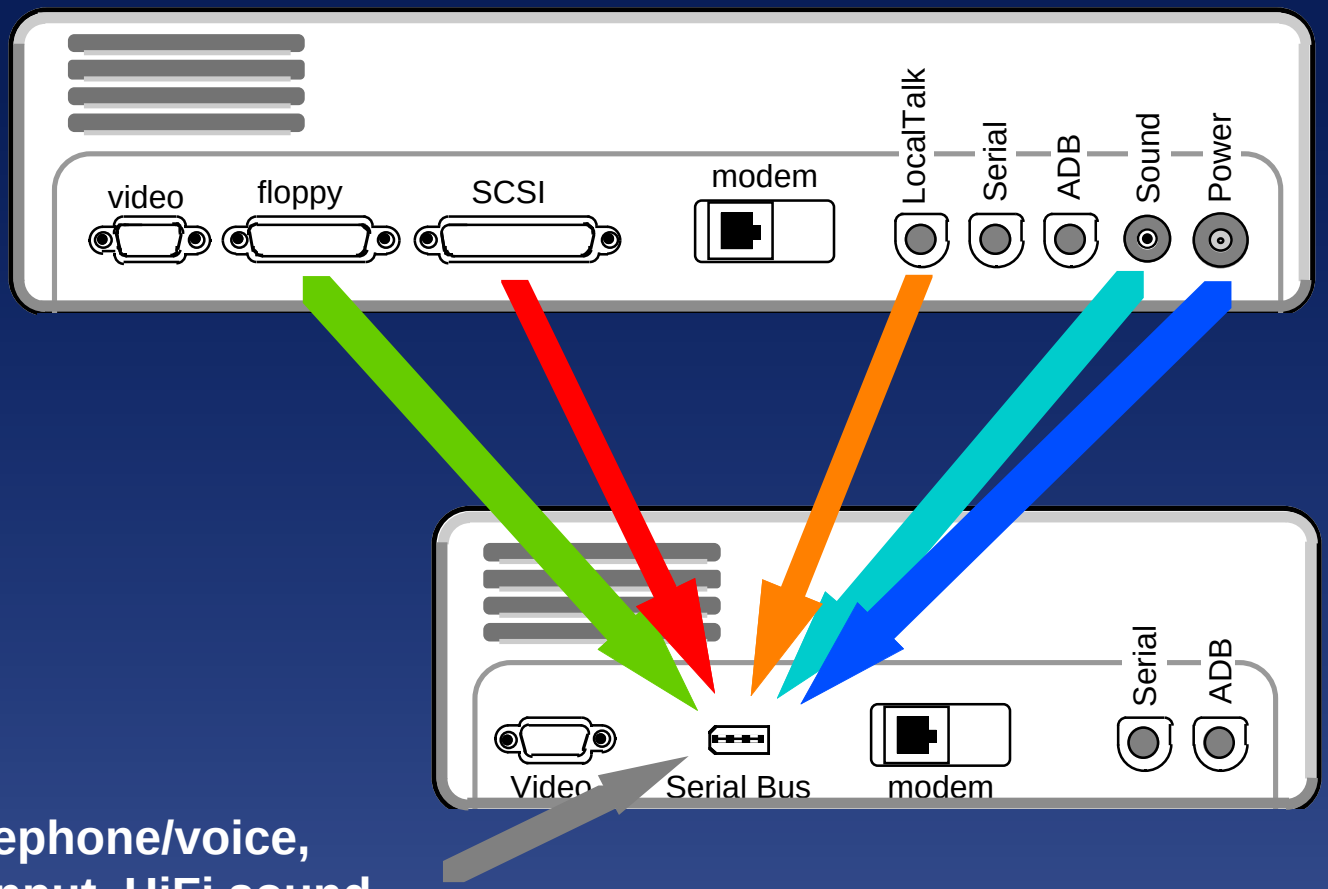


Serial Bus is *"unsupervised cabling"* — "non-cyclic network"; automatic address selection, no terminators, locations are arbitrary

Data paths (peer-to-peer)



Clean up the desktop cable mess!

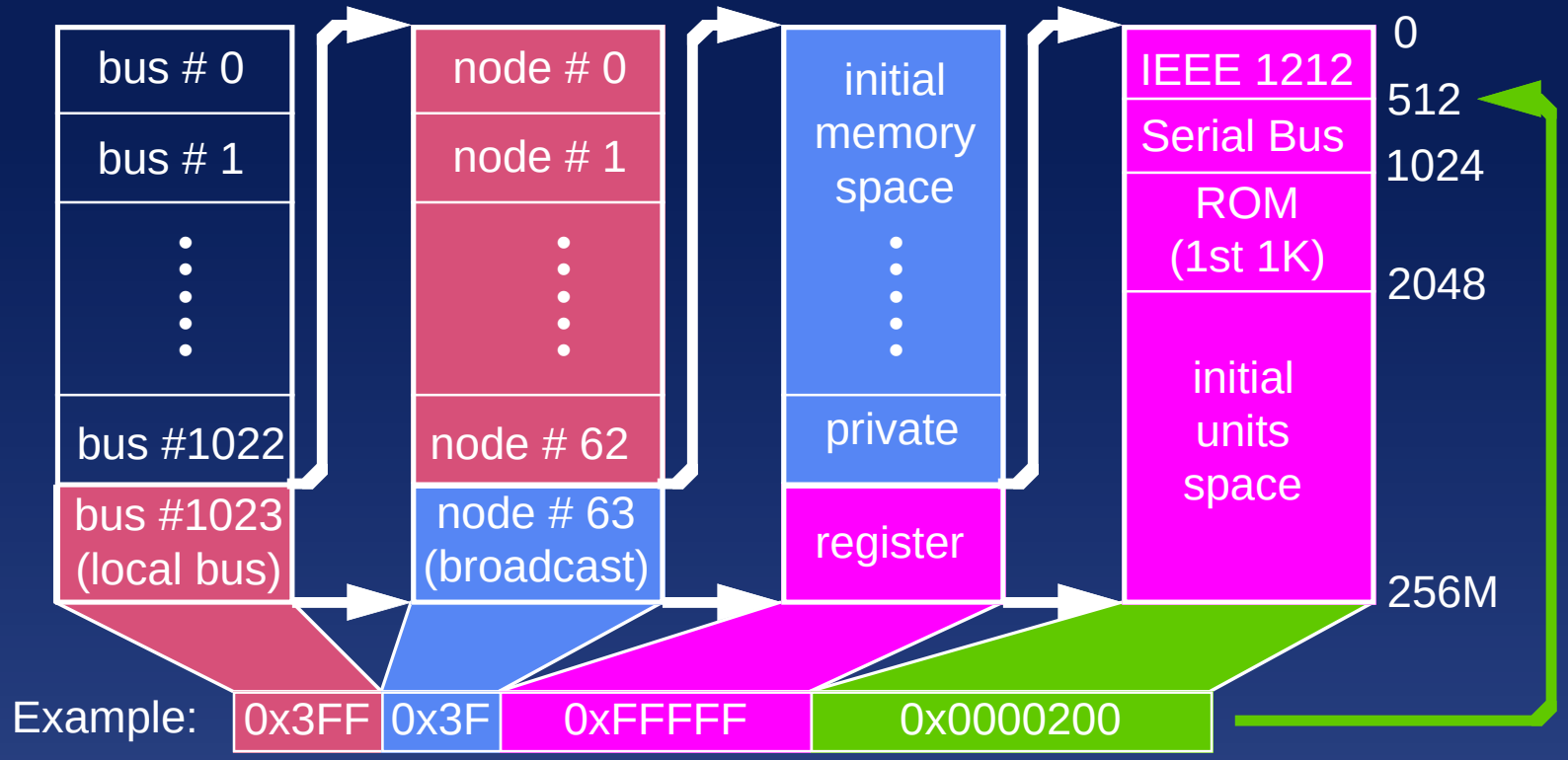


plus telephone/voice, sound input, HiFi sound, compressed video

Protocols

- IEEE 1394-1995 High Speed Serial Bus
  - “Memory-bus-like” logical architecture
  - Serial implementation of 1212 architecture
- IEEE 1212-1991 CSR Architecture
  - Standardized addressing
  - Well-defined control and status registers
  - Standardized transactions
- X3T10 Serial Bus Protocol and IEEE P1285
  - Integration of DMA into I/O process
  - SBP is based on SCSI-3 CDB structures
  - IEEE P1285 treats disk as software-managed slow cache

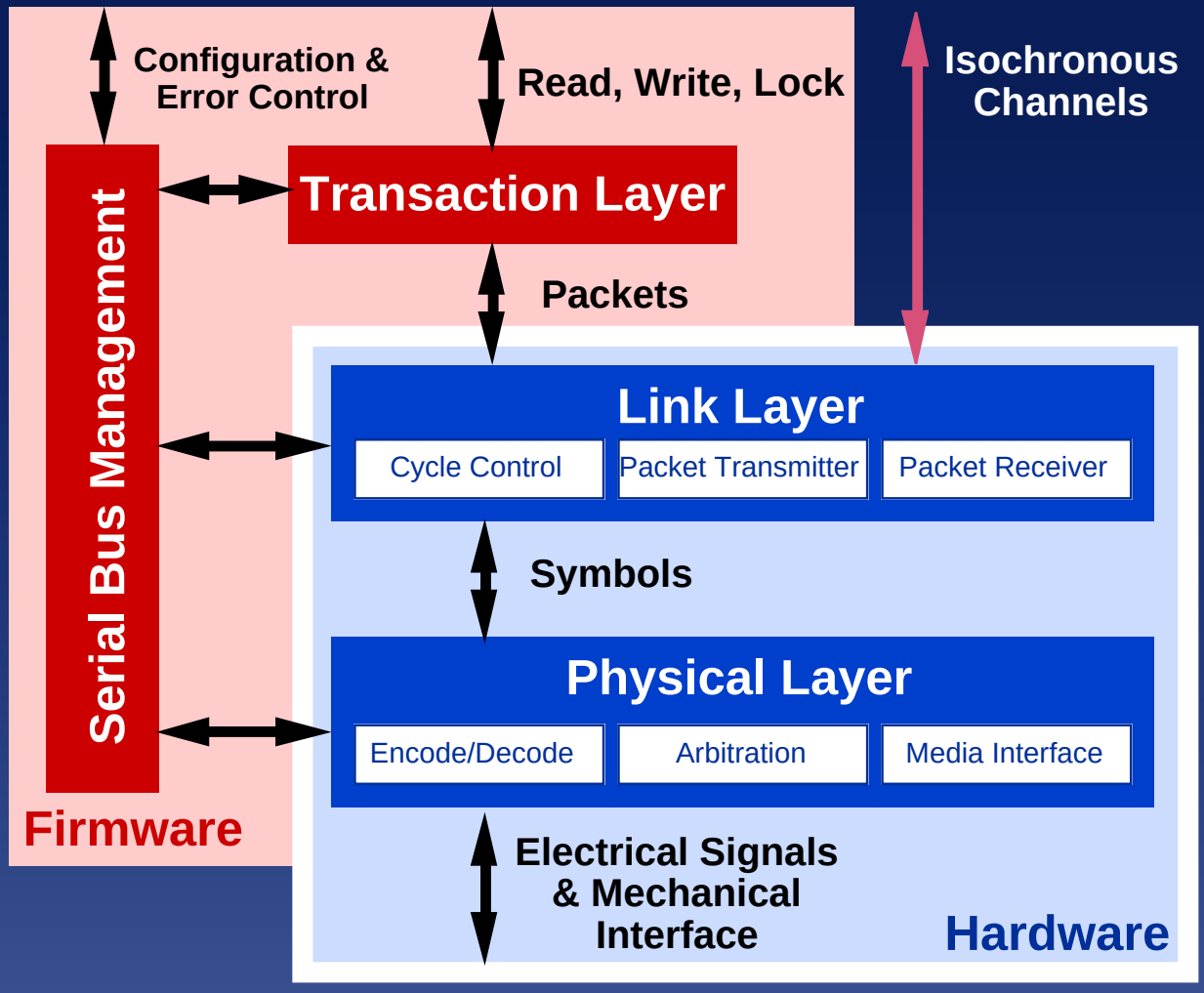
IEEE 1212 addressing



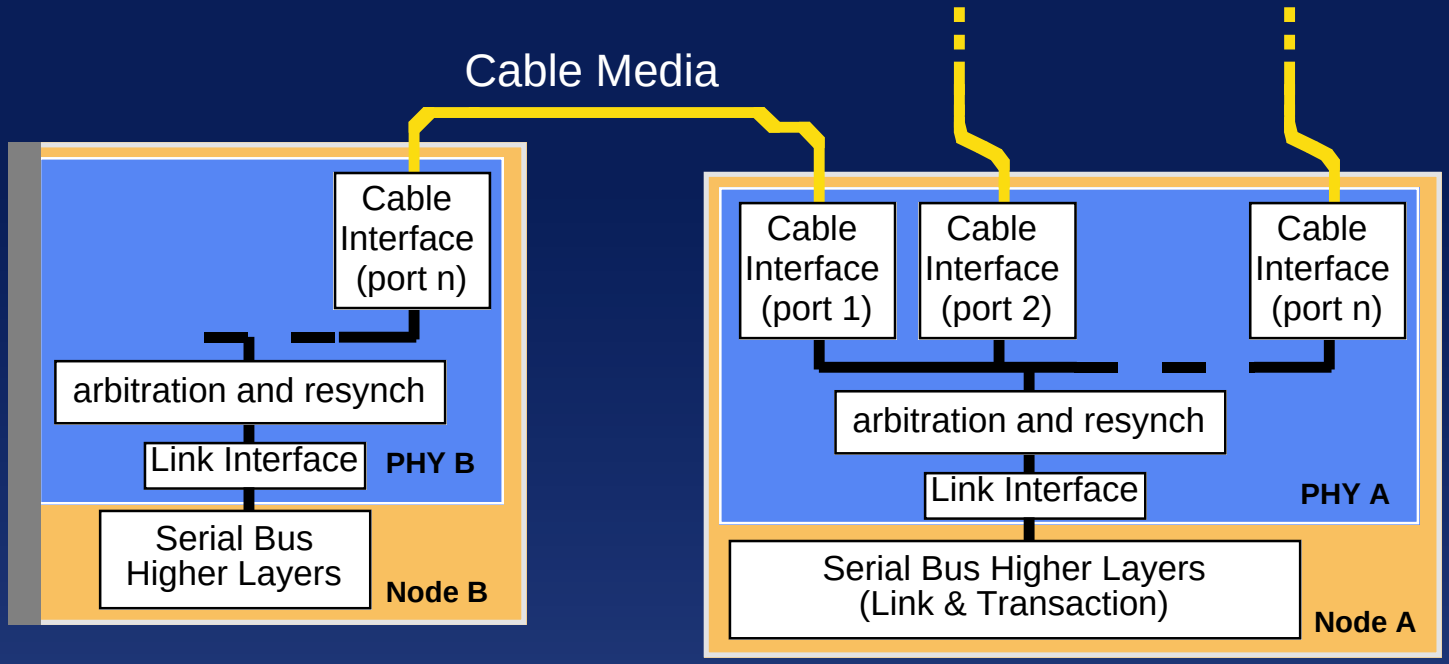
= all cycle timer registers on local bus

- **The serial bus uses “64-bit fixed” addressing**

IEEE 1394 protocol Stack



Cable interface

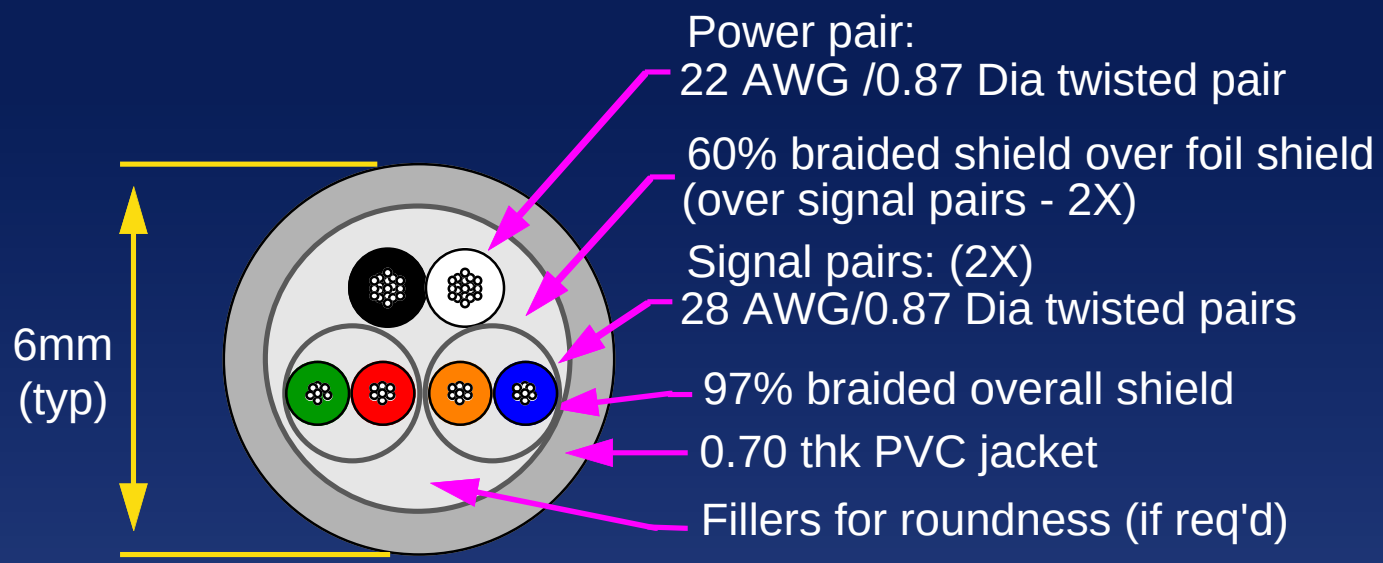


- PHY transforms point-to-point cable links into a logical bus
- Cables and transceivers are bus repeaters
- Baseline limit of 4.5 m between nodes
  - farther possible with thicker cables or <400 Mbit/sec

Cable media

- 3-pair shielded cable
  - Two pairs for data transport
  - One pair for peripheral power
- Small and rugged connector
  - Two sockets in the same area as one mini-DIN socket
- CMOS transceiver
  - 220 mv differential
  - 4 ma drive
  - < 500 ps worst case jitter

Cable media example



- **Capable of operation at 400 Mbit/sec for 4.5 m**
- **Slightly thicker wire allows 10 meter operation**

Cable interface features

- Live attach/detach
  - System protected from power on/off cycling
  - Higher layers provide simple management

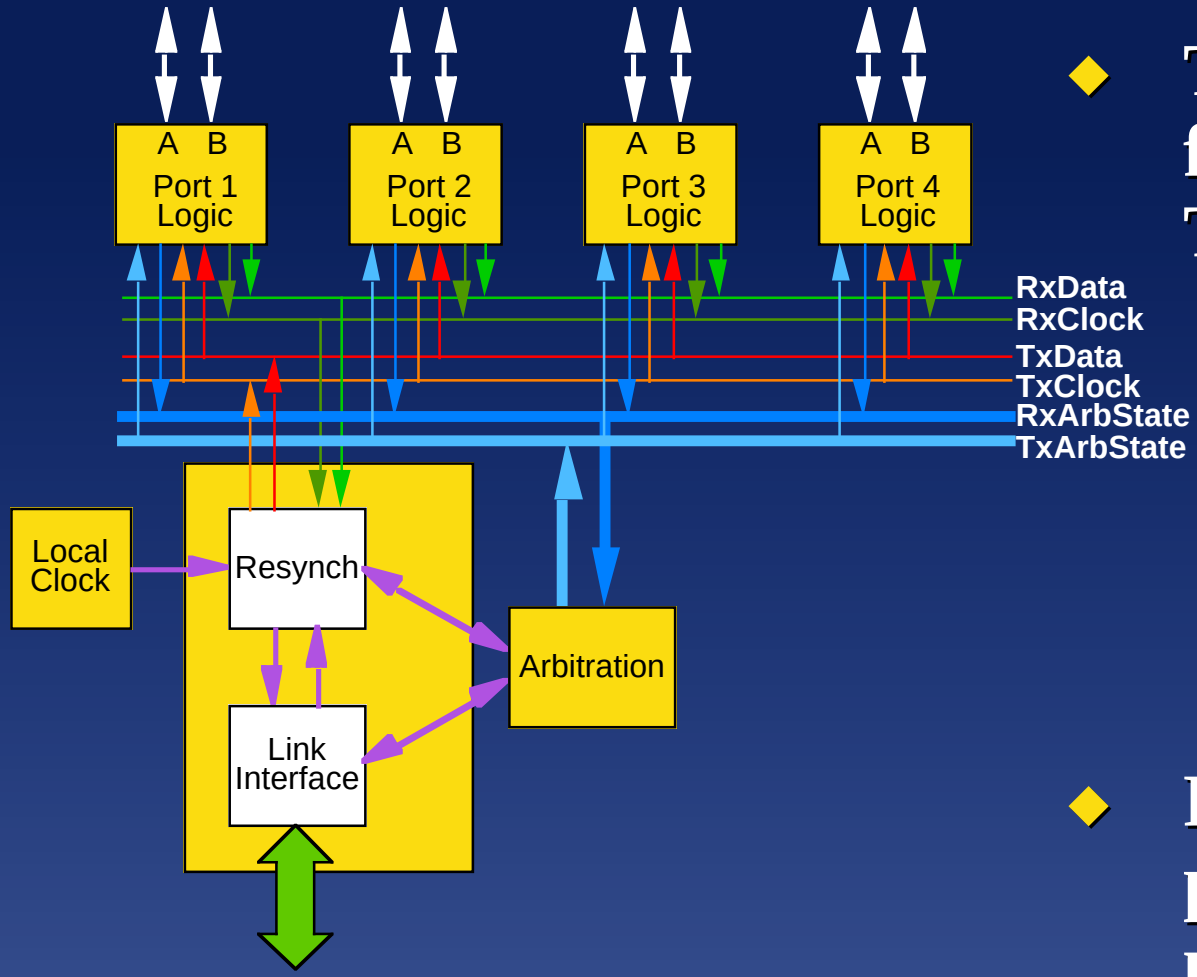
Peripheral power

- 8-40 VDC carried by cable
- Total available power is system dependent
  - Node power requirements must be declared in configuration ROM
- Cable system allows up to 1.5 A (60 watts) per link
  - Nodes can either source or sink power
  - Multiple power sources on one bus provide additional flexibility

Physical layer

- 98.304 Mbit/sec half duplex transport
  - Data reclocked at each node
  - 196.608, 393.216, ... Mbit/sec growth paths
- Data encoding
  - Data and strobe on separate pairs
  - Automatic speed detection
- Fair and priority access
  - Tree-based handshake arbitration
  - Automatic assignment of addresses

Example cable PHY IC

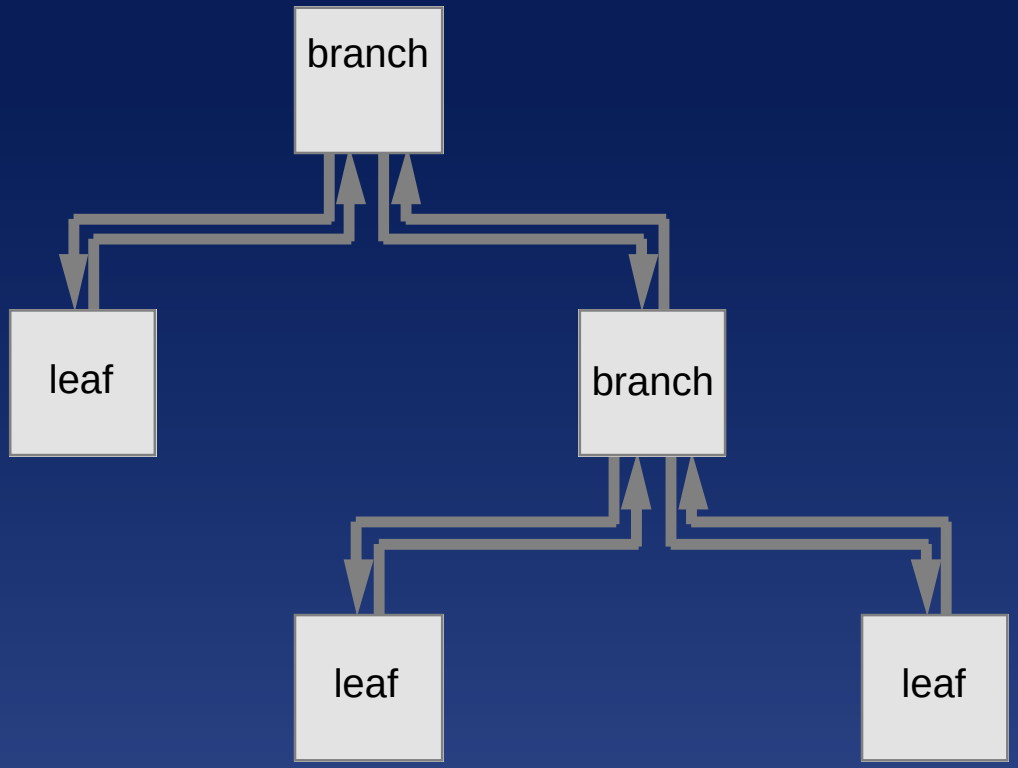


- Two twisted pairs for data: TPA and TPB
  - TPA is transmit strobe, receive data
  - TPB is receive strobe, transmit data
  - Both are bidirectional signals, both are used in arbitration
- Reclocks repeated packet data signals using local clock

Cable arbitration phases

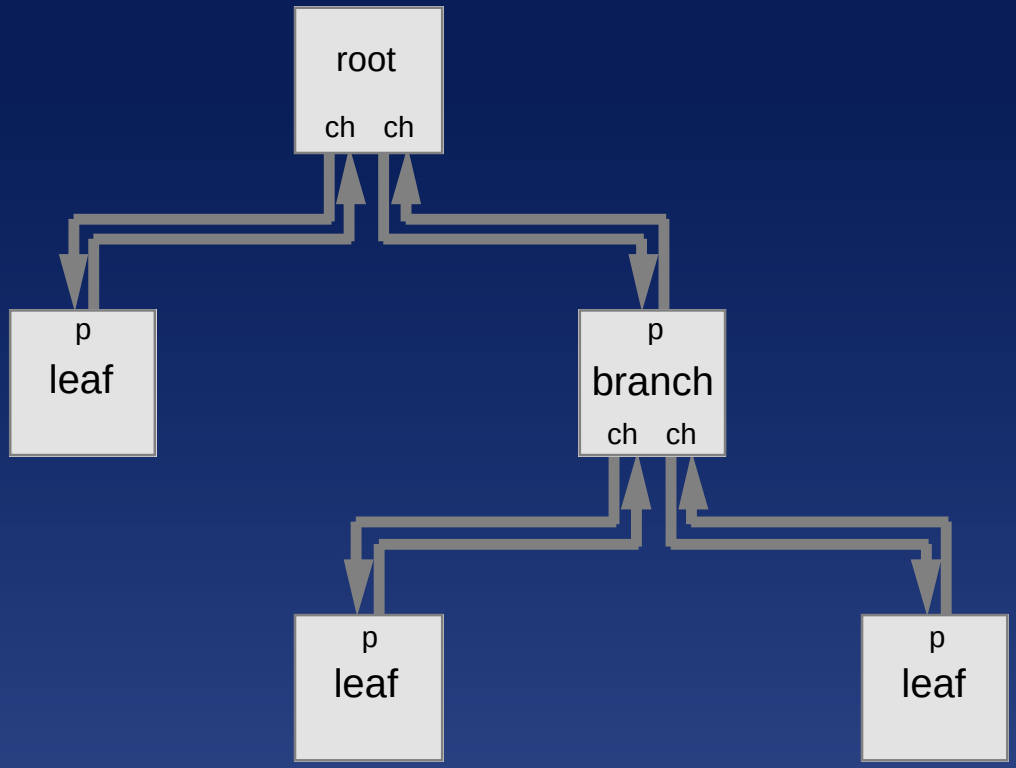
- **Reset**
  - Used whenever reconfiguration needed
  - Live insertion & new cycle master are examples
- **Tree Identification**
  - Transforms a simple net topology into a tree
- **Self Identification**
  - Assigns physical node number (Node ID)
  - Exchange speed capabilities with neighbors
- **Normal Arbitration**
  - Root has highest priority

Tree identification #1



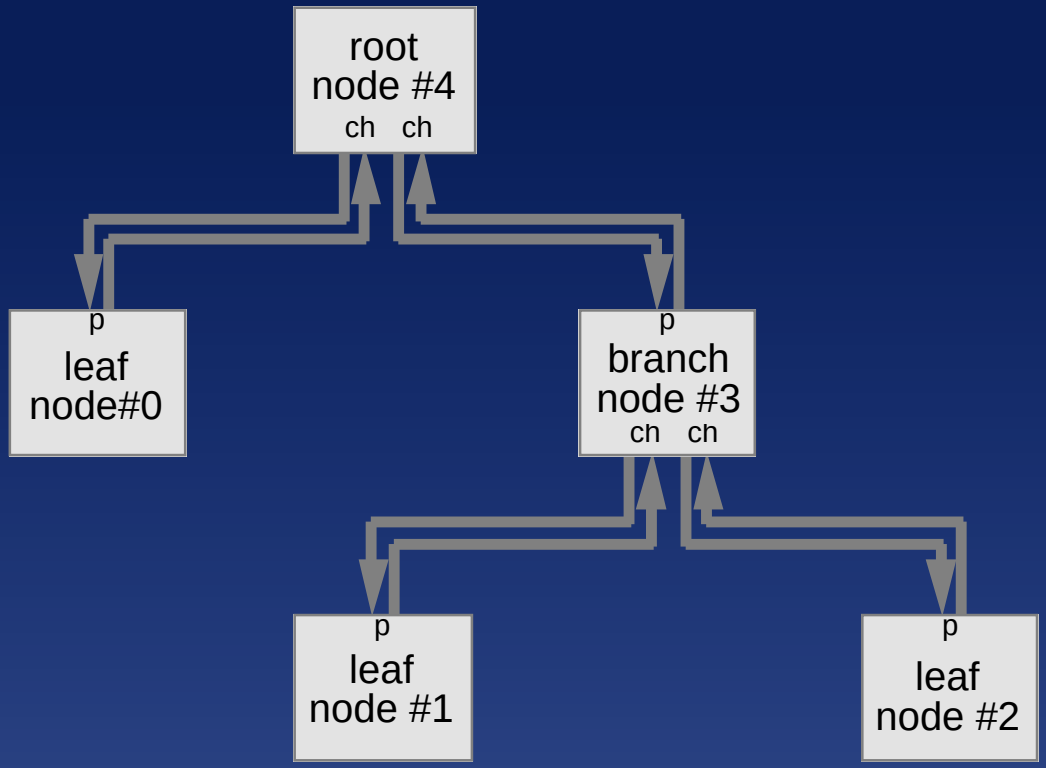
- After reset, each node only knows if it is a leaf (one connected port) or a branch (more than one connected port)

Tree identification #2



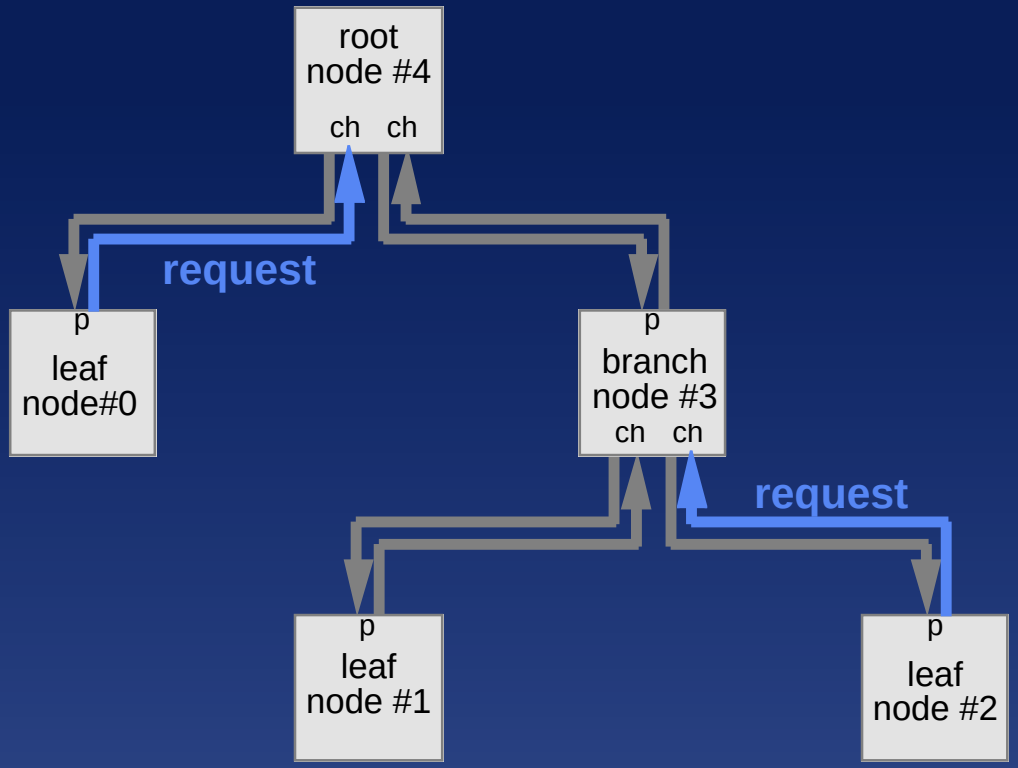
- After Tree ID process, the Root node is determined and each port is labeled as pointing to a child or a parent

Self identification



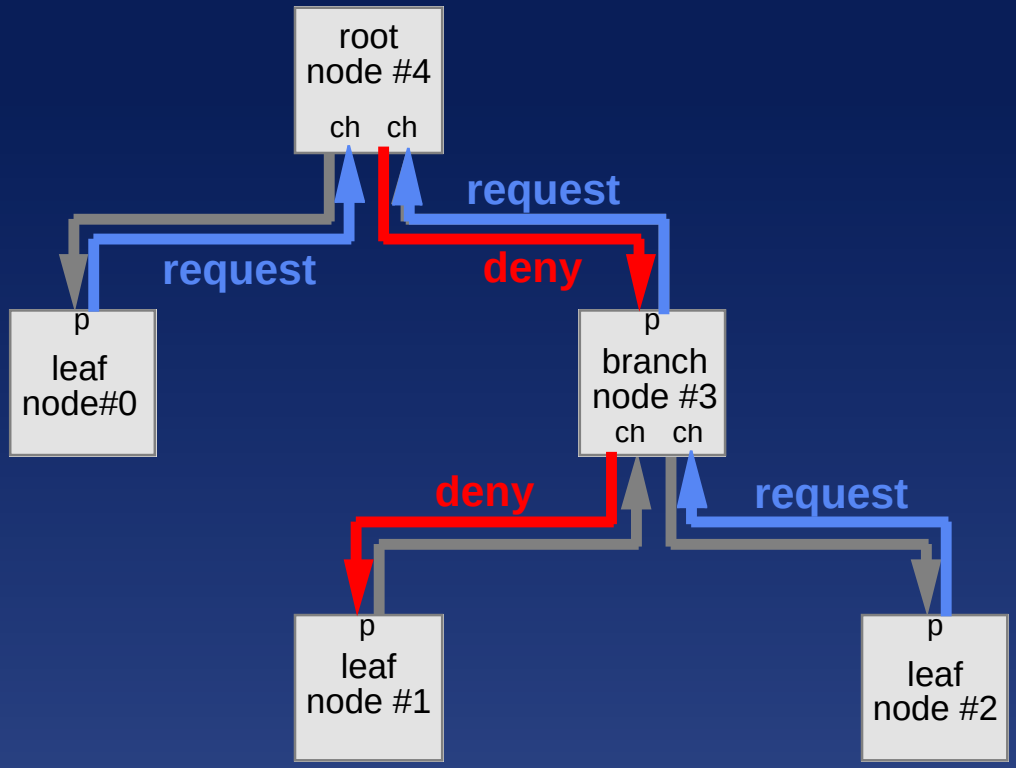
- **After the self ID process, each node has a unique physical node number, and the topology has been broadcast**

Normal arbitration #1



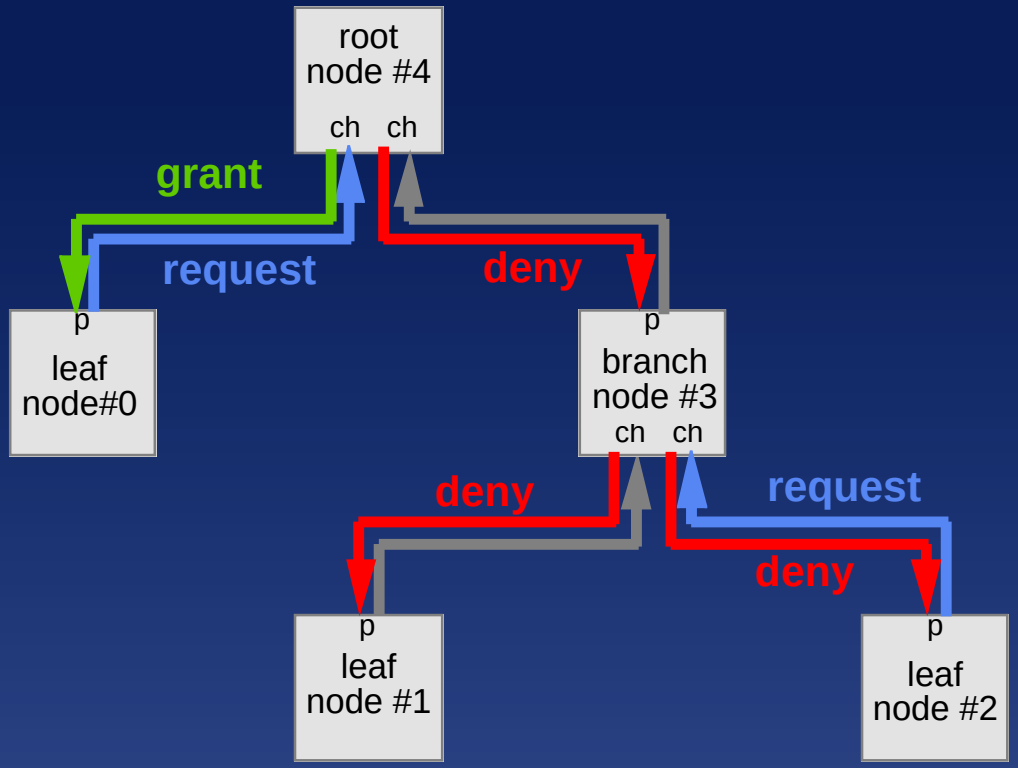
- **Suppose nodes #0 and #2 start to arbitrate at the same time, they both send a request to their parent ...**

Normal arbitration #2



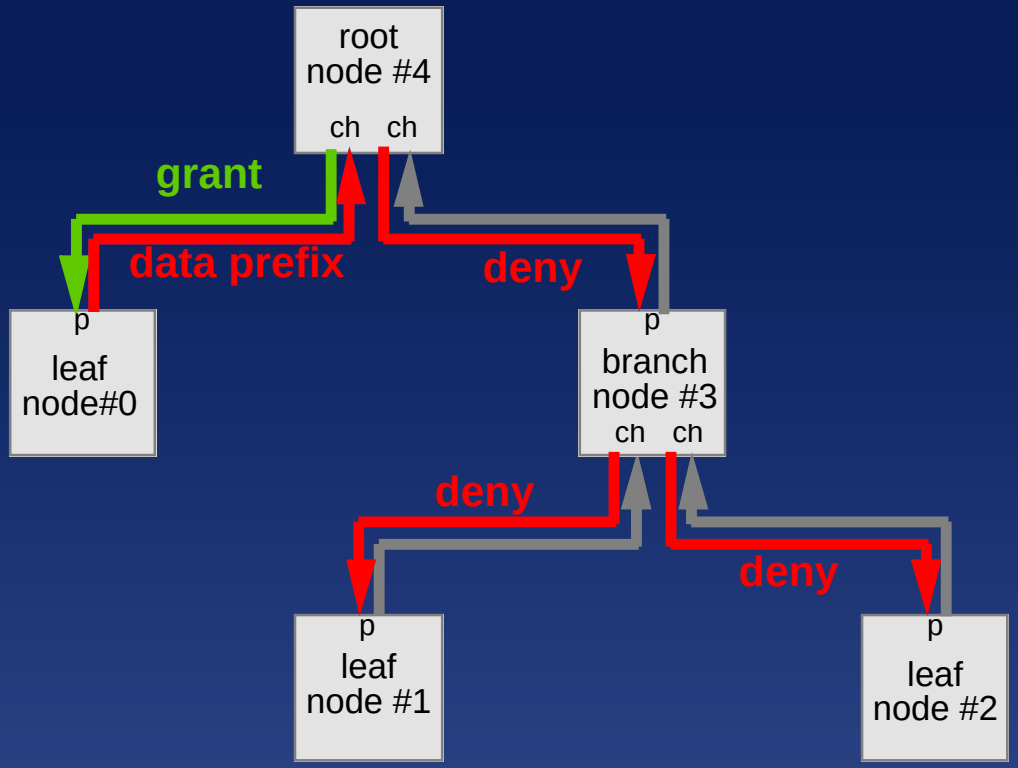
- **The parents forward the request to their parent and deny access to their other children ...**

Normal arbitration #3



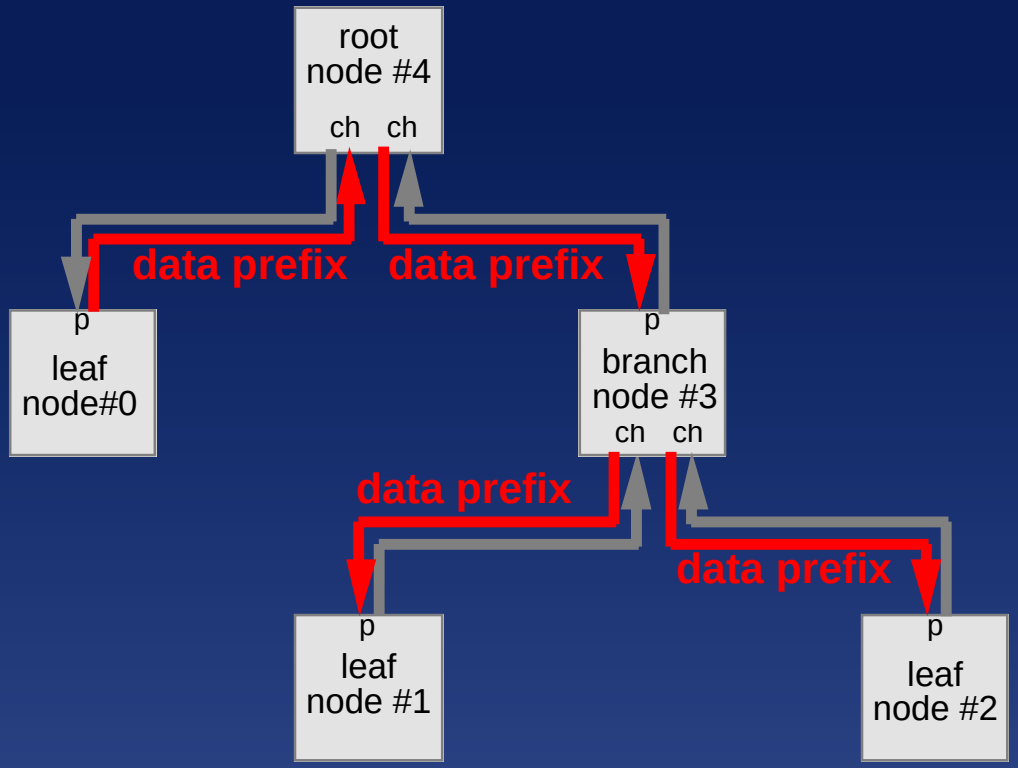
- The root grants access to the first request (#0), and the other parent withdraws it's request and passes on the deny ...

Normal arbitration #4



- The winning node #0 changes its request to a data transfer prefix, while the loosing node #2 withdraws its request ...

Normal arbitration #5



- **The parent of node 1 sees the data prefix and withdraws the grant, and now all nodes are correctly oriented to repeat the packet data (a "deny" is a "data prefix!) ...**

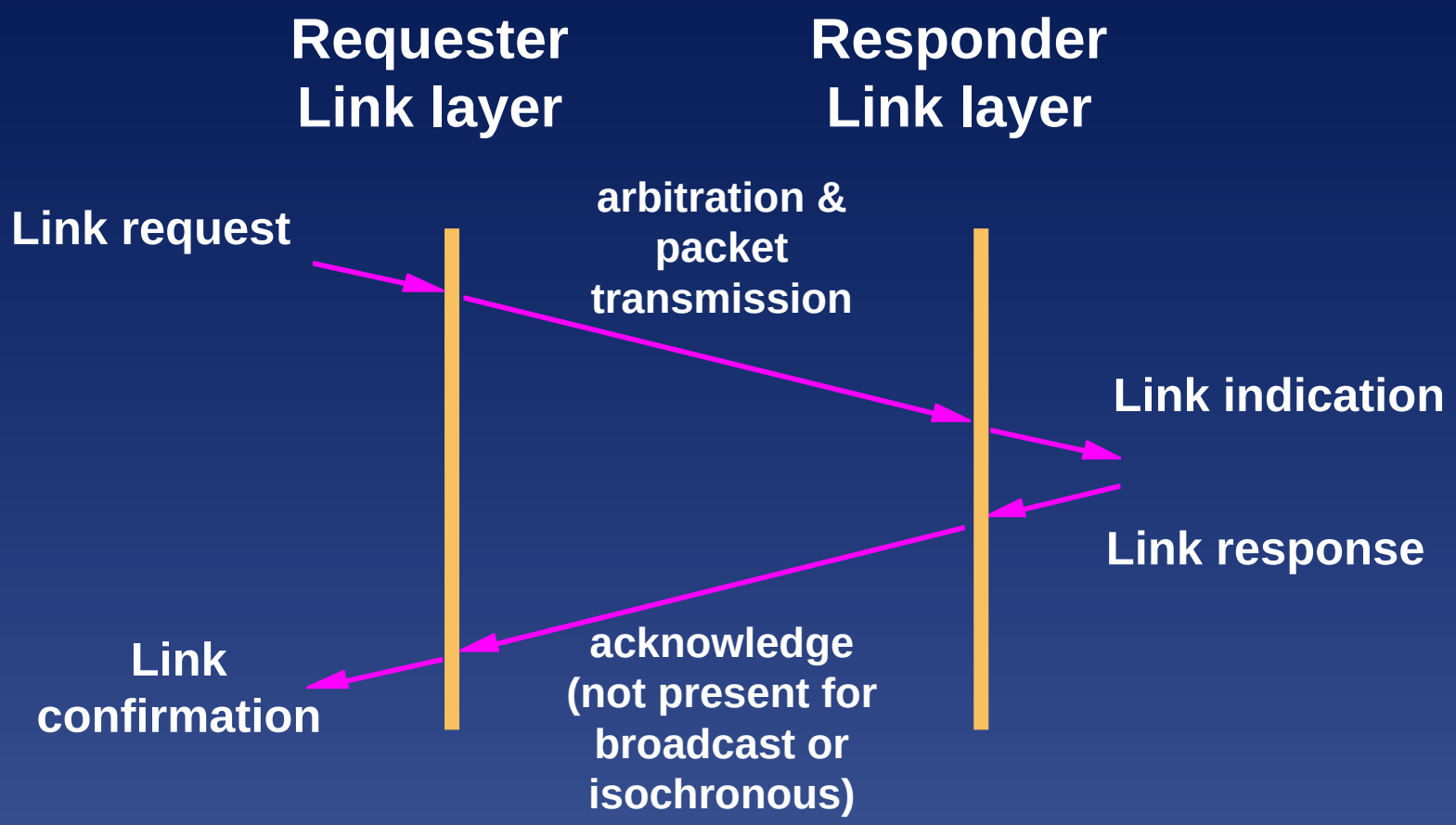
Link layer

- Implements acknowledged datagram service
  - Called a "subaction" of arbitration, packet transmission, and acknowledge
- Flexible addressing using 1212 architecture
  - Direct 64-bit addressing (48 bits per node)
  - Hierarchical addressing for up to 63 nodes on 1023 busses

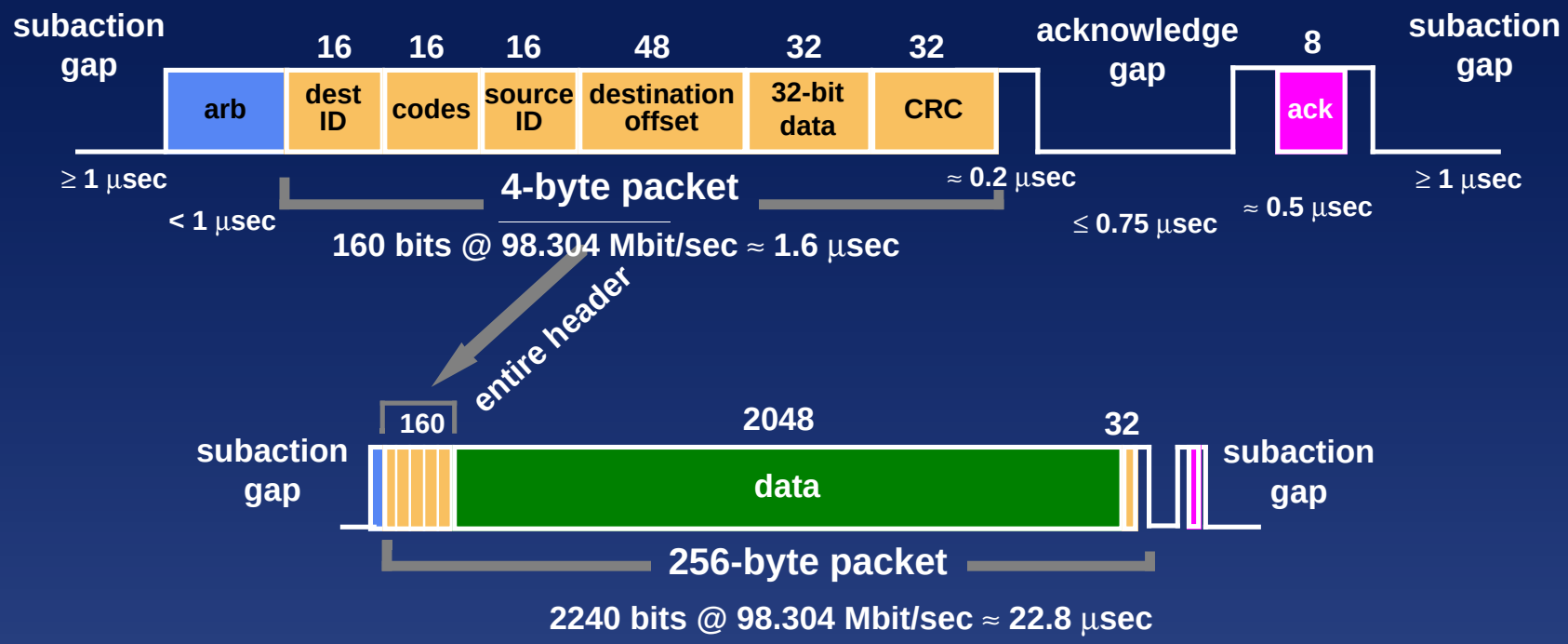
Isochronous transport

- Optional
  - But required for multimedia applications
- Multiple "channels" each 125 μsec "cycle" period
  - Channel count limited by available bandwidth
- Variable channel size up to ≈1000 bytes/cycle
  - Up to ≈2000 bytes/cycle at 196 Mbit/sec

Link layer operation

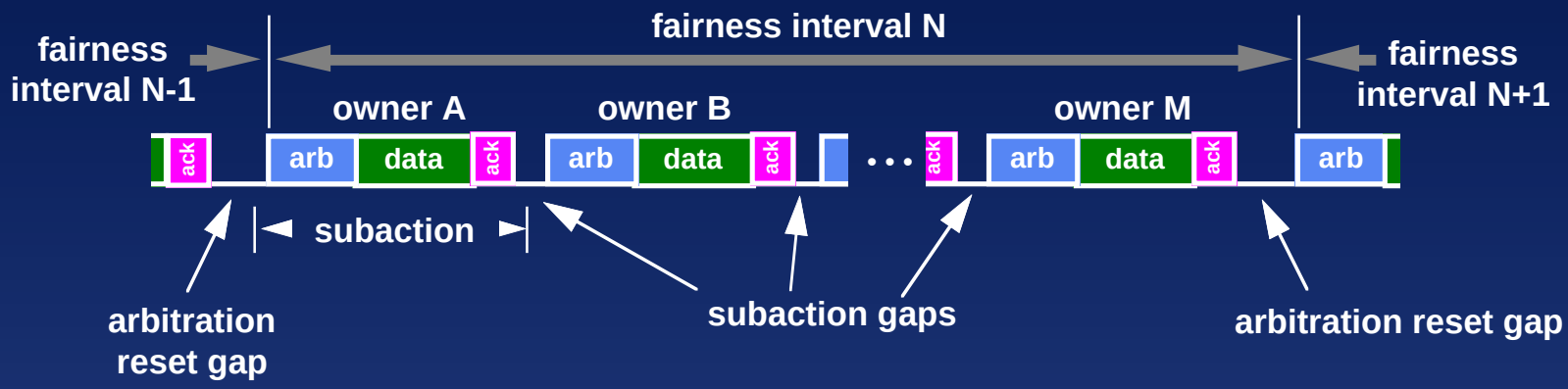


Example packets



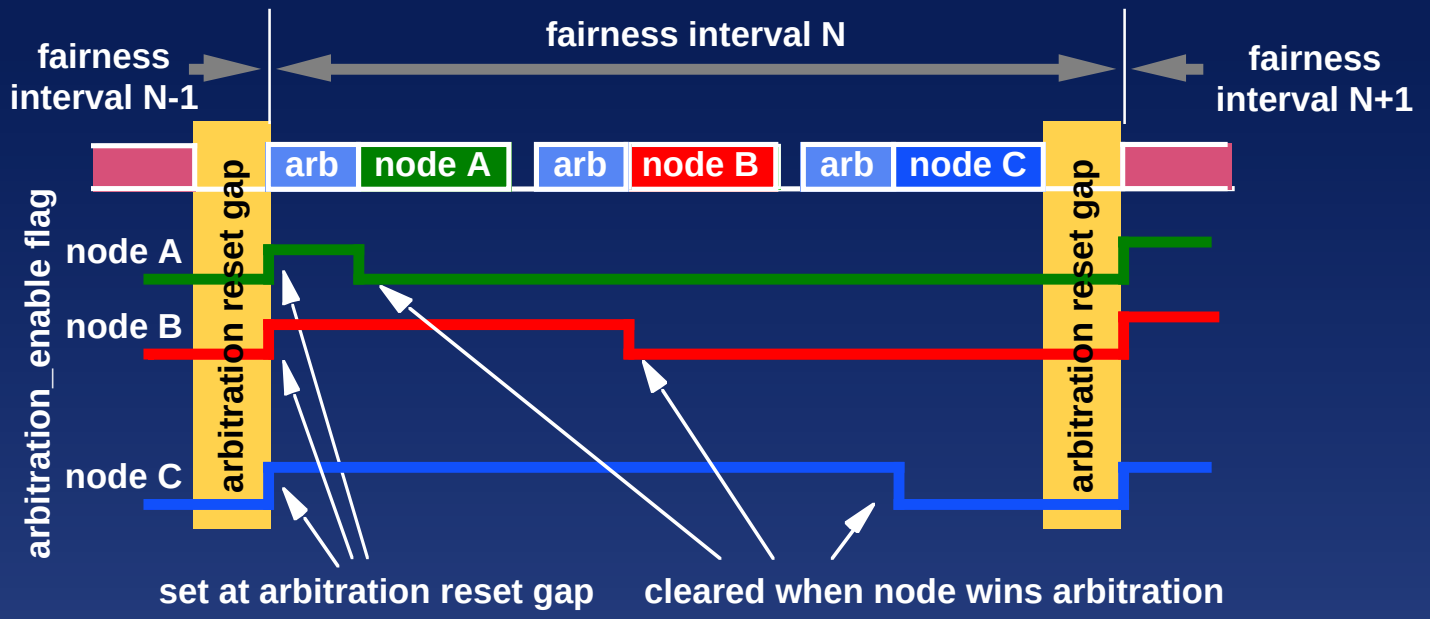
- **Actual efficiency very good**
  - **10 Mbyte/sec information throughput including all of the "SBP" SCSI-3 protocol using 100 Mbit/sec rate (~80%)**

Fairness interval



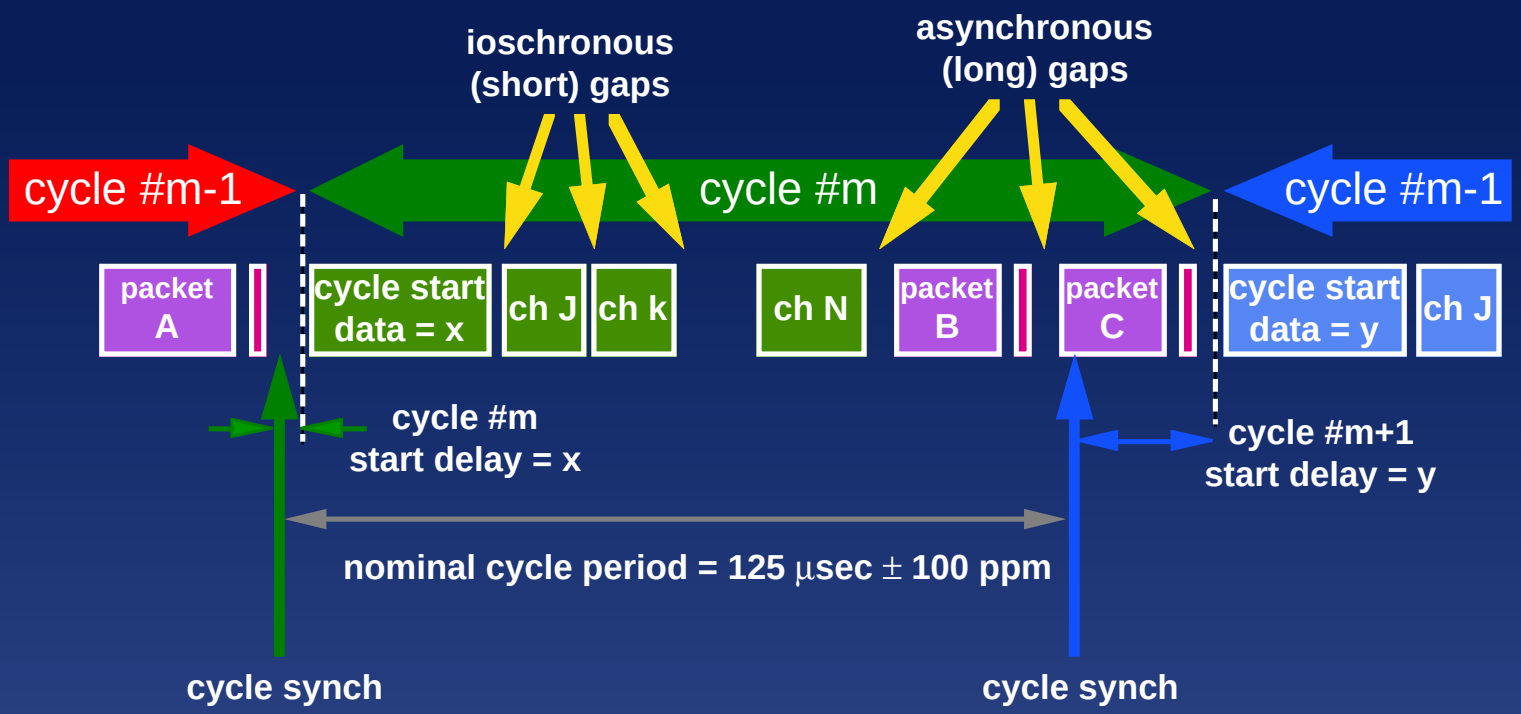
- Fairness Interval is bounded by “arbitration reset gaps”
- Reset gaps are longer than normal subaction gaps

Fair arbitration



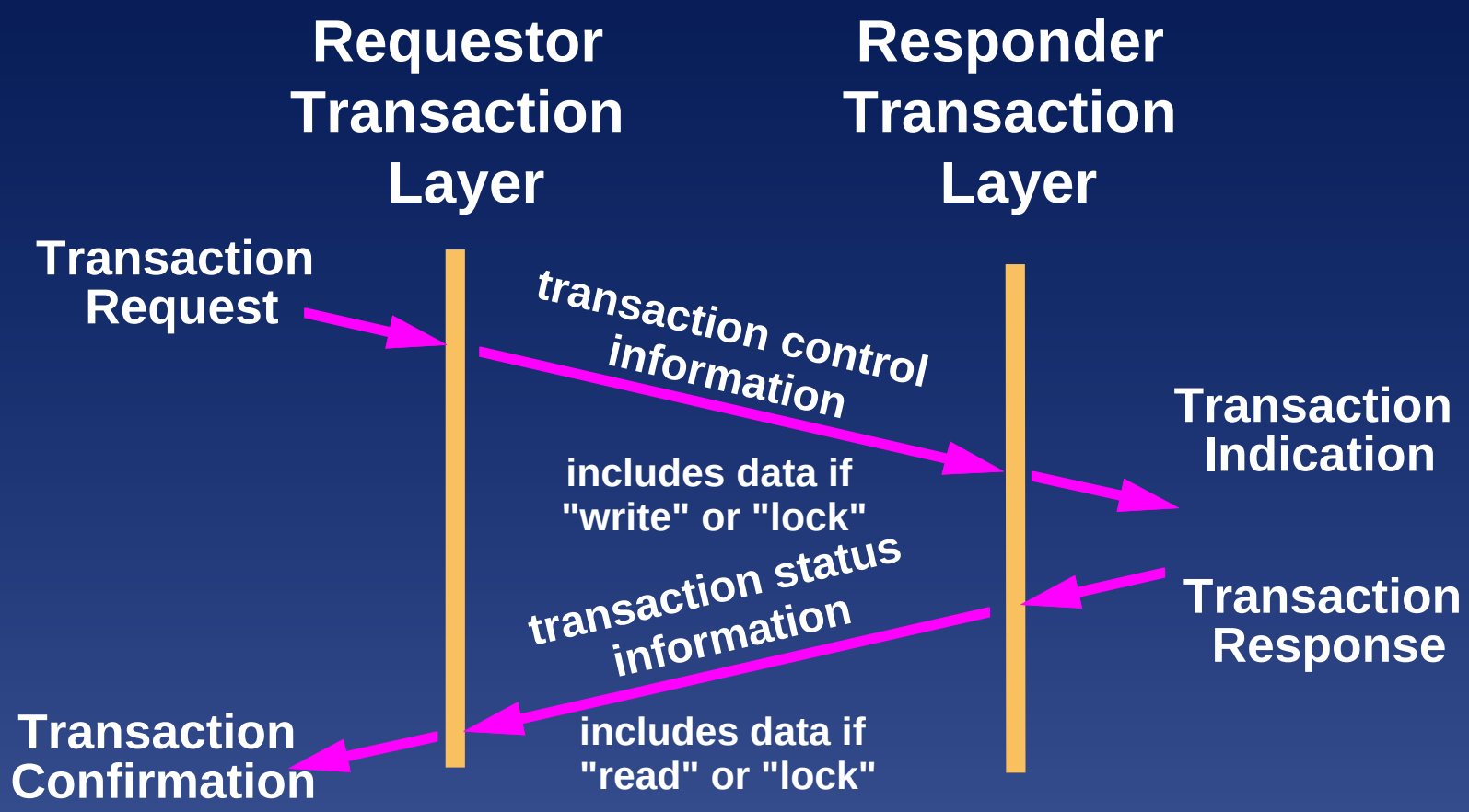
- Each node gets one access opportunity each Fairness Interval
  - special case for isochronous data

Cycle structure



- Scott Smyers of Sony to elaborate

Transaction layer



Multiple transaction types

- Simplified 4-byte (quadlet) read and write are required
- Variable-length block read and write are optional
- Lock transactions optional
  - Swap, Compare-and-swap needed for bus management

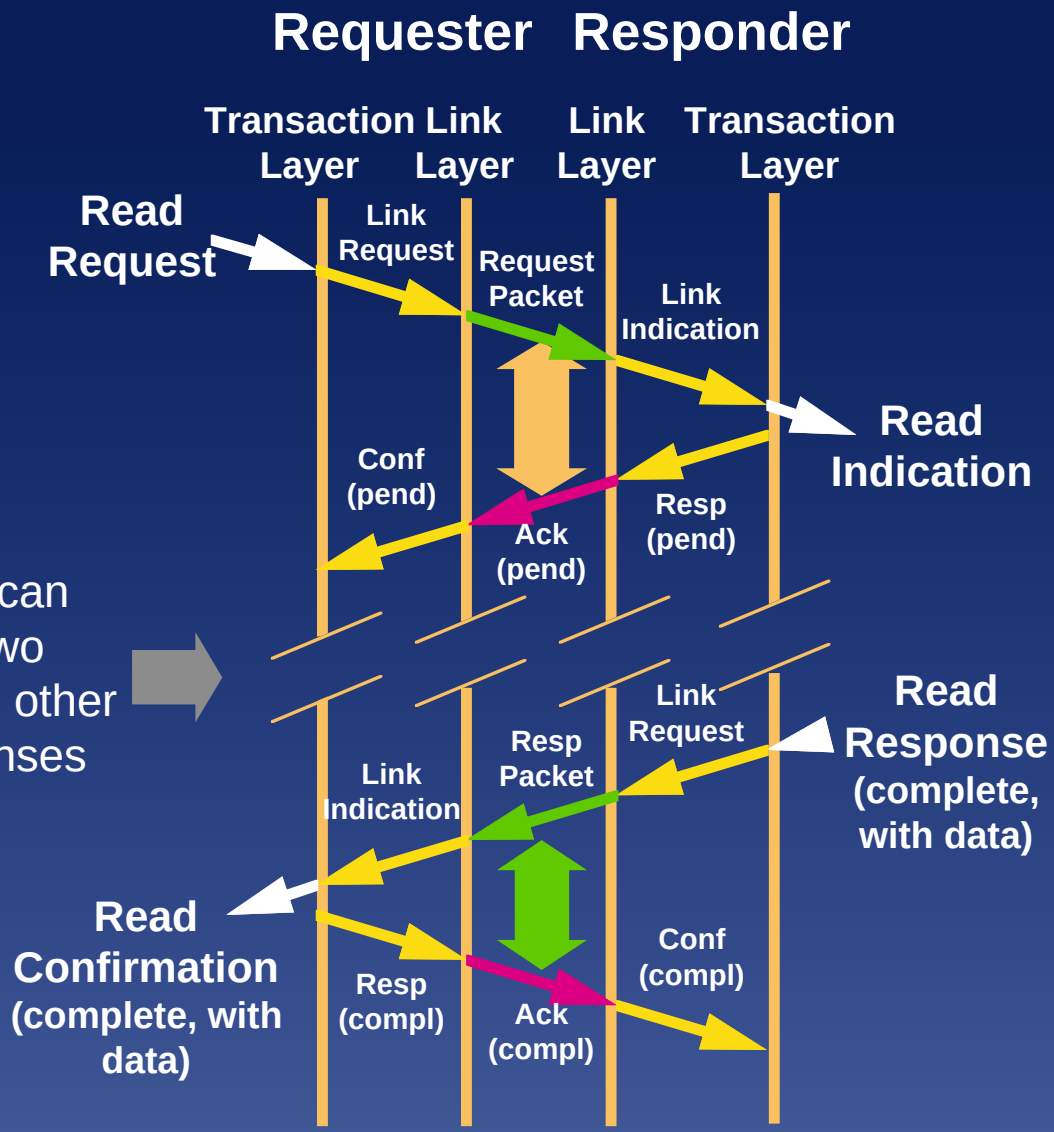
Efficient media usage

- Split transactions required
  - Transactions have request and response parts
  - Bus is never busy unless data is actually being transferred
- Request and response can be unified two ways
  - "Read" and "Lock" can have concatenated subactions
  - "Write" can have immediate completion

Split transaction

- **Request and response have separate subactions**

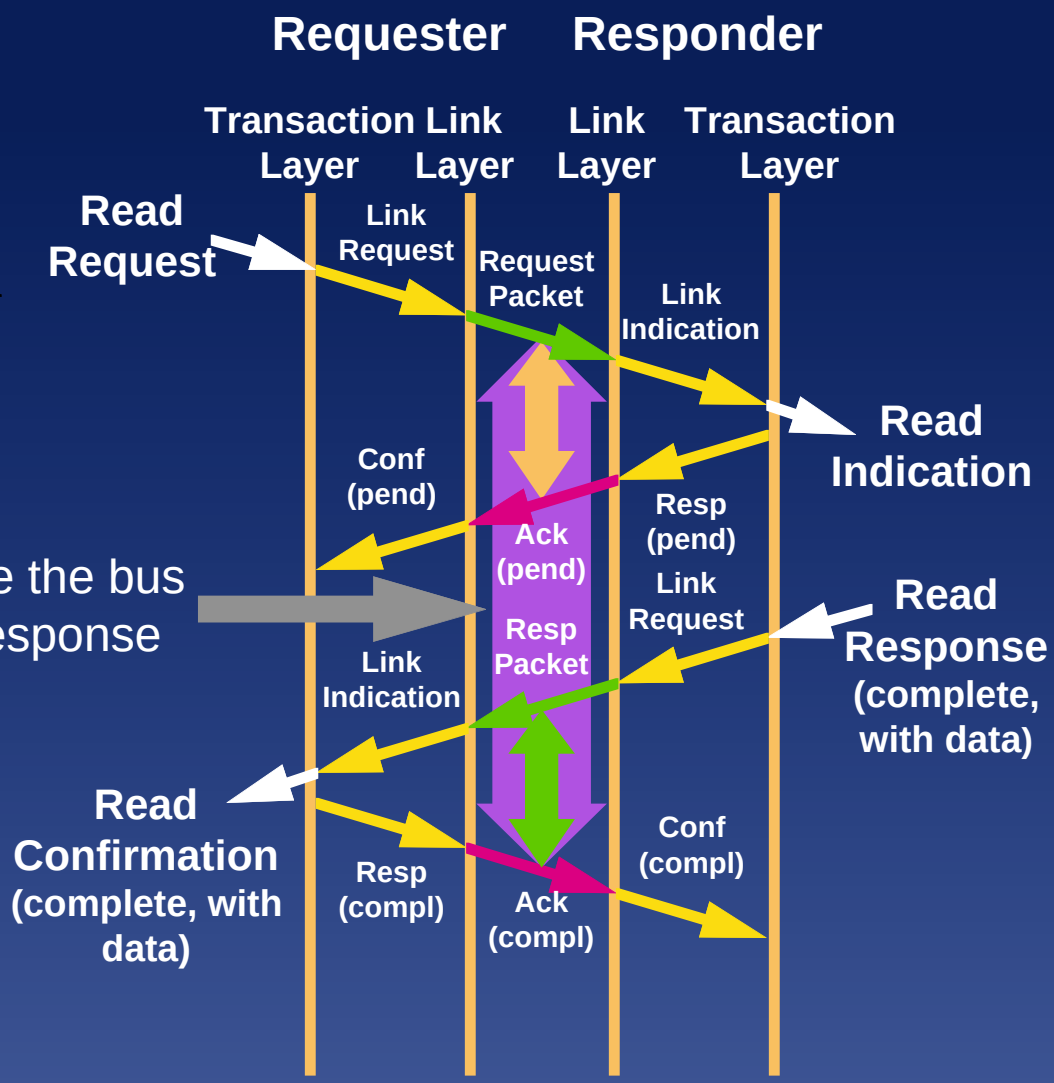
Other Link-Layer operations can take place between these two subactions, ***including*** sending other transaction requests or responses



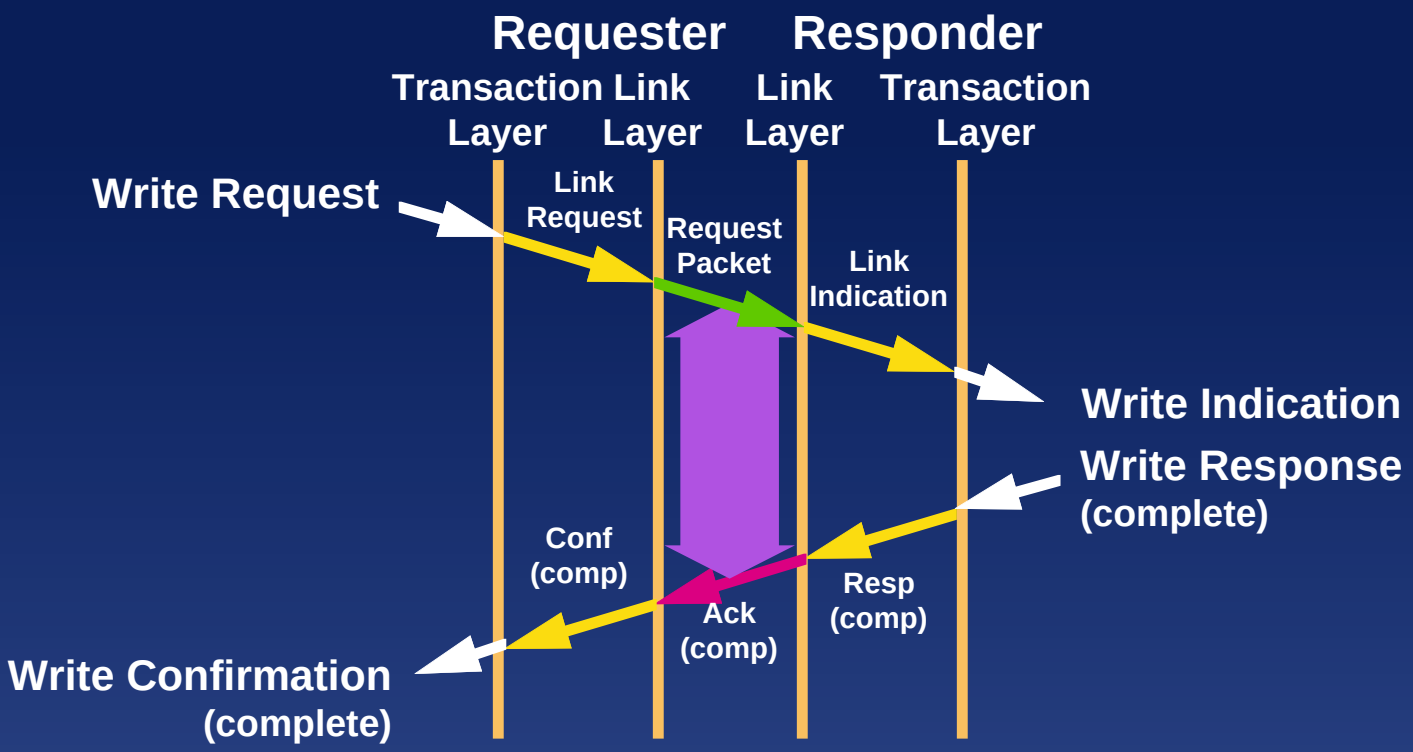
Concatenated transaction

- **Used if responder is fast enough to return data before ack is completed**

the responder does not release the bus after sending the ack, sends response packet within 1.5µsec



Unified transaction



- Only used for write transactions

Bus management

- Automatic address assignment
  - Done in physical layer
- Standardized addresses and configuration ROM from IEEE 1212 architecture
- Resource management
  - Isochronous channels and bandwidth
  - Power
- Peter Johanssen of Congruent to elaborate

Futures

- Gigabit rates/longer distances
  - 1-8 Gbit/sec, 25+ meter hops
- Incremental addition of nodes without bus reset
- Advanced power management
- Bridging issues
  - for > 63 devices, or for isolation of high-bandwidth local traffic

How does 1394 help?

- Much better human interface
  - smaller, more rugged connectors with defined usage
  - Hot plugging, no manual configuration
- Excellent real performance
  - High true data rates
  - Direct map to processor I/O model
  - DMA is simple: CPU memory directly available to peripherals
    - examples: SBP and IEEE 1285 support direct scatter/gather buffers

... but even more important

- It's inexpensive
  - For computers, it’s almost as cheap as single-ended 8-bit SCSI
    - will be cheaper since it's silicon-intensive
  - Much less expensive for peripherals and consumer electronics
- Direct support for isochronous data
  - Likely choice for digital consumer video, high-end audio
  - Media servers get cheaper

Getting documentation

- “IEEE 1394-1995 High Performance Serial Bus”
  - IEEE Standards Office +1-908-981-1393
- Internet
  - ftp://ftp.apple.com/pub/standards/p1394
  - Drafts in FrameMaker 5 or Acrobat "pdf" format
  - Papers, email digests, performance reports
- Internet email reflector
  - “P1394@Sun.COM”
  - Administrator is “Bob.Snively@Eng.Sun.COM”
- 1394 Trade Association
  - http://www.firewire.org